

Discrete Event (“Network”)
Modeling, Patient Flow & Irregular
Geometries in AnyLogic

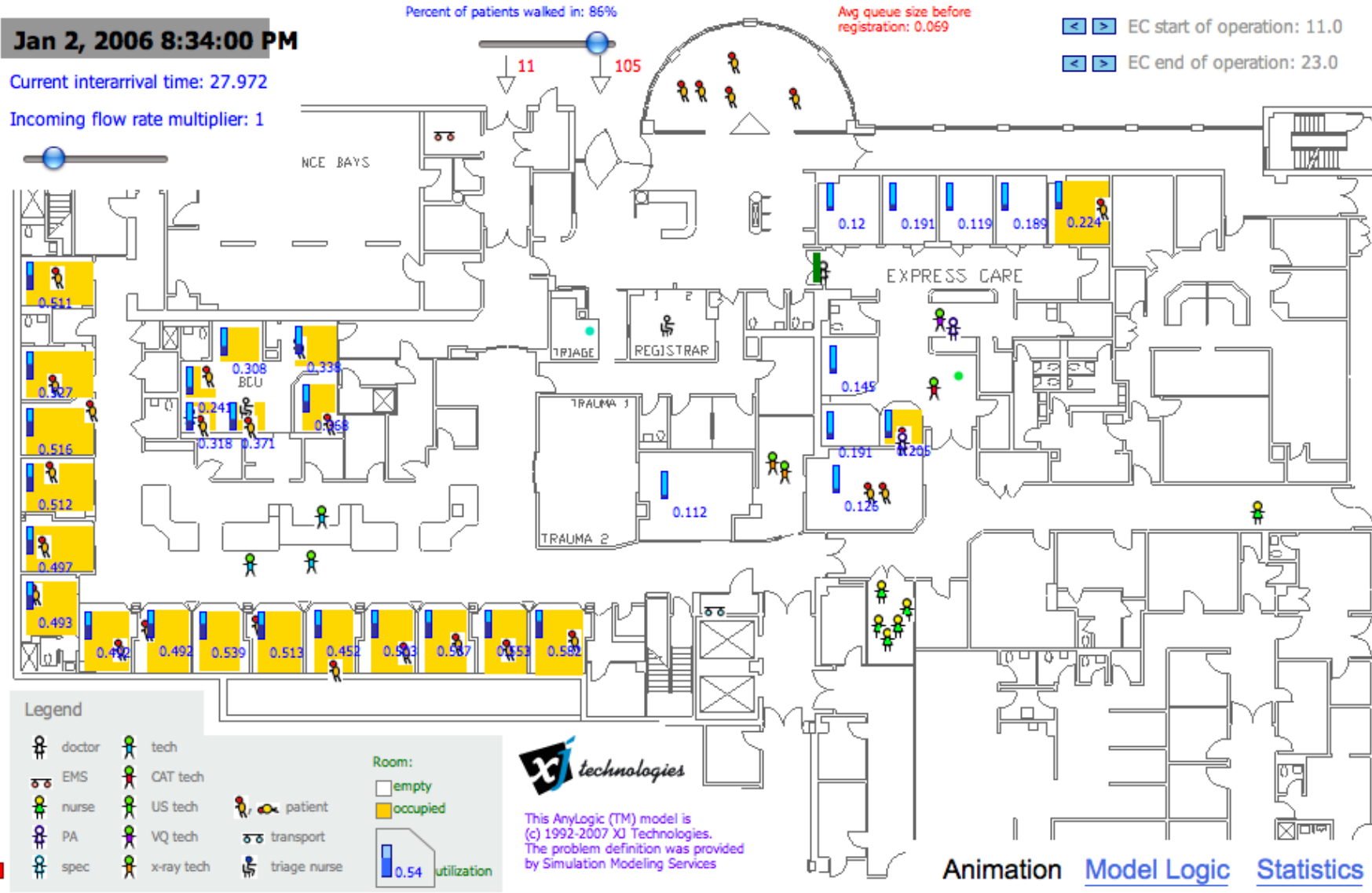
Nathaniel Osgood

MIT 15.879

April 6, 2012

Recall: "Network Modeling" Irregular Spatial Embedding

Emergency Department





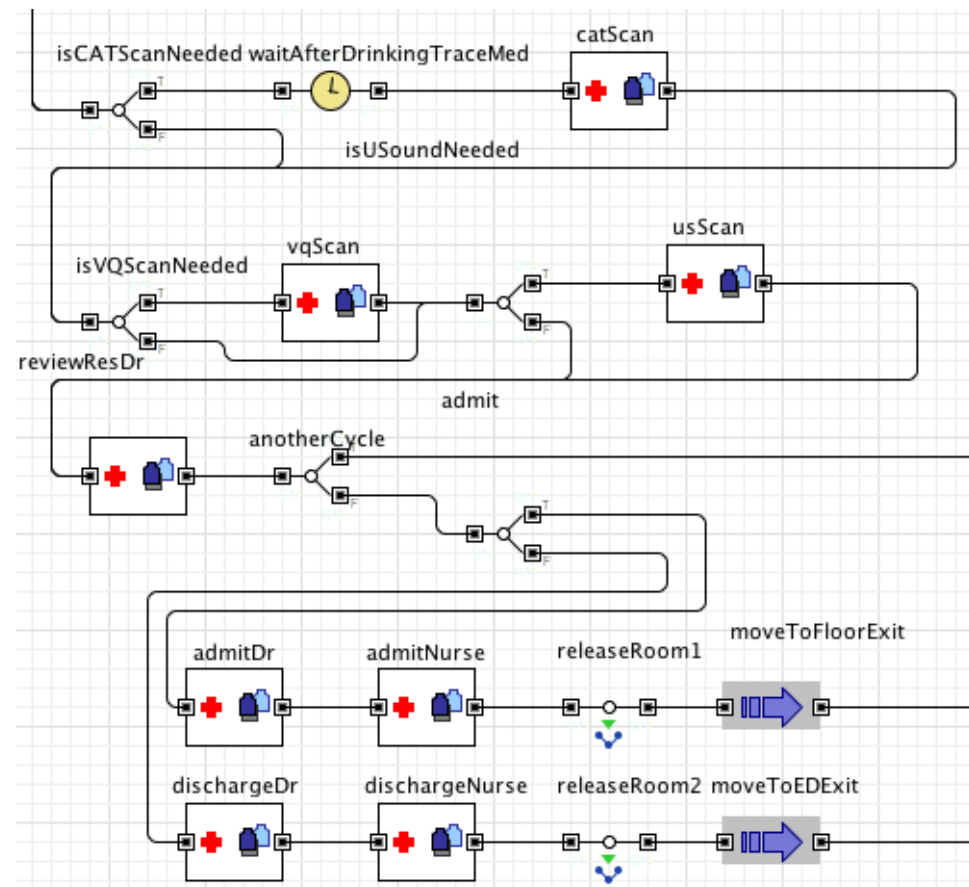
Hands on Model Use Ahead



Load model (available on STELLAR Site):
Ophthalmology Department.alp

Discrete Event (“Network Oriented”) Modeling

- Resource-based modeling
 - Queues
 - Processes
 - Flow charts
 - Capacitated resource pools
 - Send to
 - Attachment/detachment



Central Concepts in Discrete Event Modeling

- Entities flowing through processes & being processed at successive stages
- Flow charts specify process (“workflow” on entities)
- Resources required for processing
 - Queues for entities awaiting resources
 - Limited-capacity resource pools from which resources are drawn
- Entity interaction with resources
 - Attachment/detachment
 - Seizure
- Physical “homes” for resources
- Movement paths (via polylines)

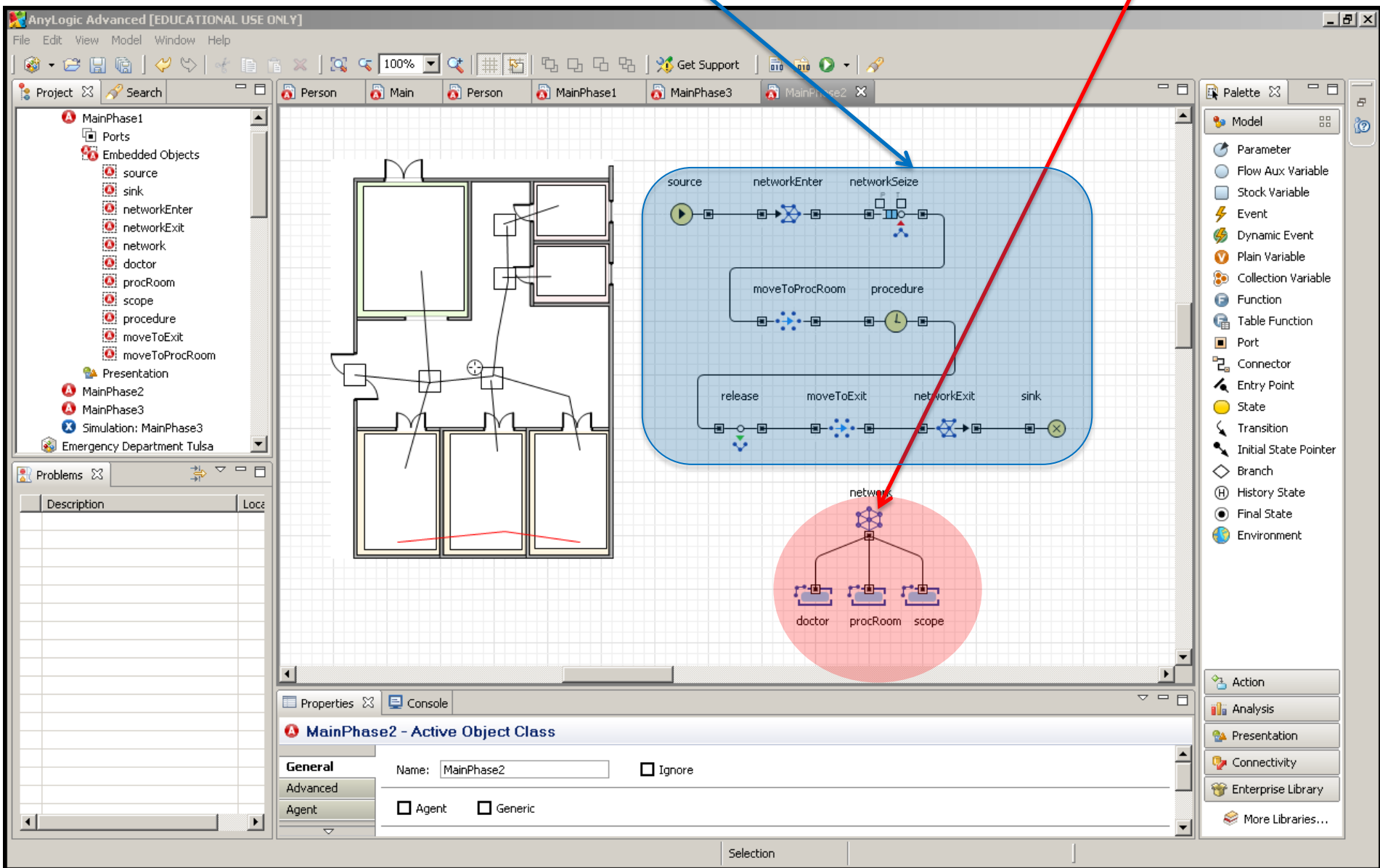
Specifying a “Network”

- A network groups together
 - Entities
 - Resources
 - Portions of the workflow

Flow chart associated with Network (Entities flow through this)

Network and Its Components

Types of resources associated with the network, each in a resource pool



Entities

- Entities are the central parties on which the processes take place
 - eg Patients in a hospital or clinic, cars in an assembly line
- Primarily passive – things “happen to them”
- “Flow through” (are routed around) the flow charts associated with the system
 - “Injected” into the system at a source; disappear at sink
 - Only exist for the duration of time that are in the system
- Multiple entities typically in the system at one time
- If wish to maintain extra information on an entity, can “subclass” the Entity class
- Entities are often associated with a physical representation, which travels around spatial enviro.

Resources

- Frequently resources are required to initiate a particular phase of processing
 - A doctor (resource) to administer surgery to a patient (entity)
 - A piece of diagnostic equipment (resource) to image a patient (entity)
 - An EKG to (resource) to record from a patient (entity)
 - A gurney or bed (resource) for a patient (entity)
- Distinctions amongst these resources
 - Portable vs. fixed
 - Mobile (with agency)
- To capture these dependencies, a network is often associated with multiple types of resources

Defining Resource Pools

Capacity of Pool (number of units of resource present)

Resource pool type
(Static [Fixed], Moving [Mobile],
Portable [Can be carried])

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network of resources (doctor, procRoom, scope) and a flow diagram. The Properties window for the 'doctor' resource pool is open, showing the following details:

- Name: doctor
- Type: NetworkResourcePool<T extends ResourceUnit>
- Package: com.xj.anylogic.libraries.enterprise
- Resource type: Moving (selected from a dropdown menu)
- Capacity defined: By home shape By table over time
- Capacity*: 5
- Speed: 10
- New resource unit: new ResourceUnit ()

A blue arrow points from the text 'Capacity of Pool' to the 'Capacity*' field in the Properties window. A red arrow points from the text 'Resource pool type' to the 'Resource type' dropdown menu in the Properties window.

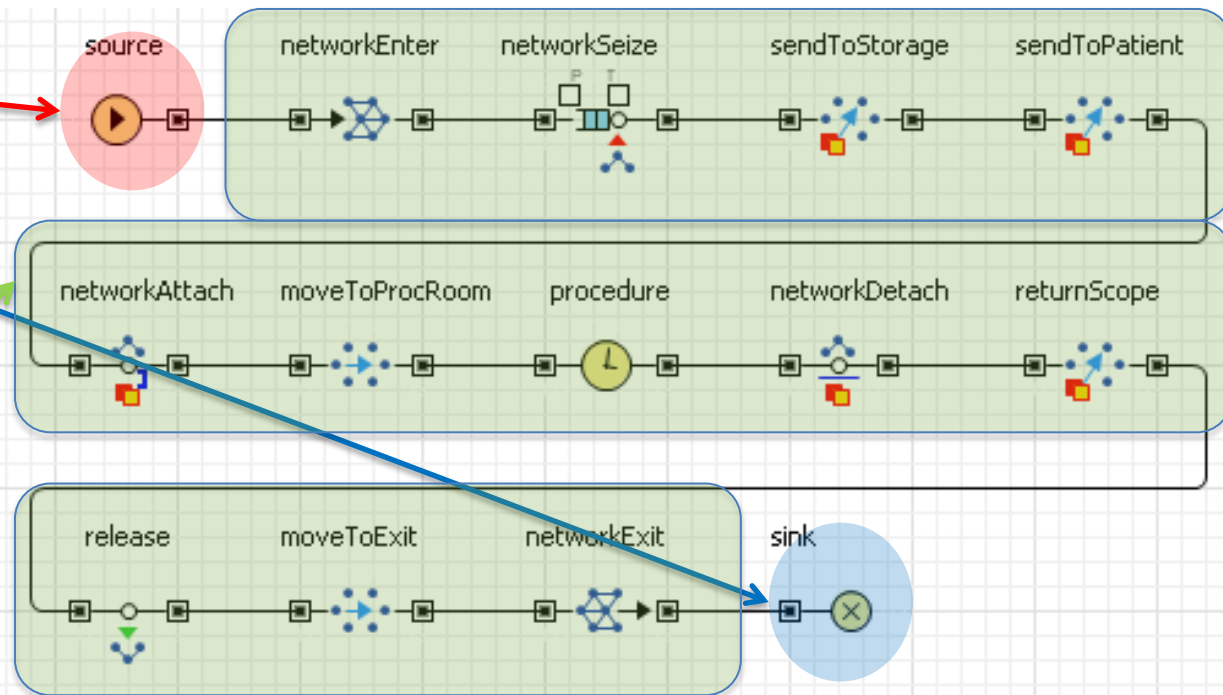
Flow Charts

- Entities flow in a single direction on flow charts

- **Start**

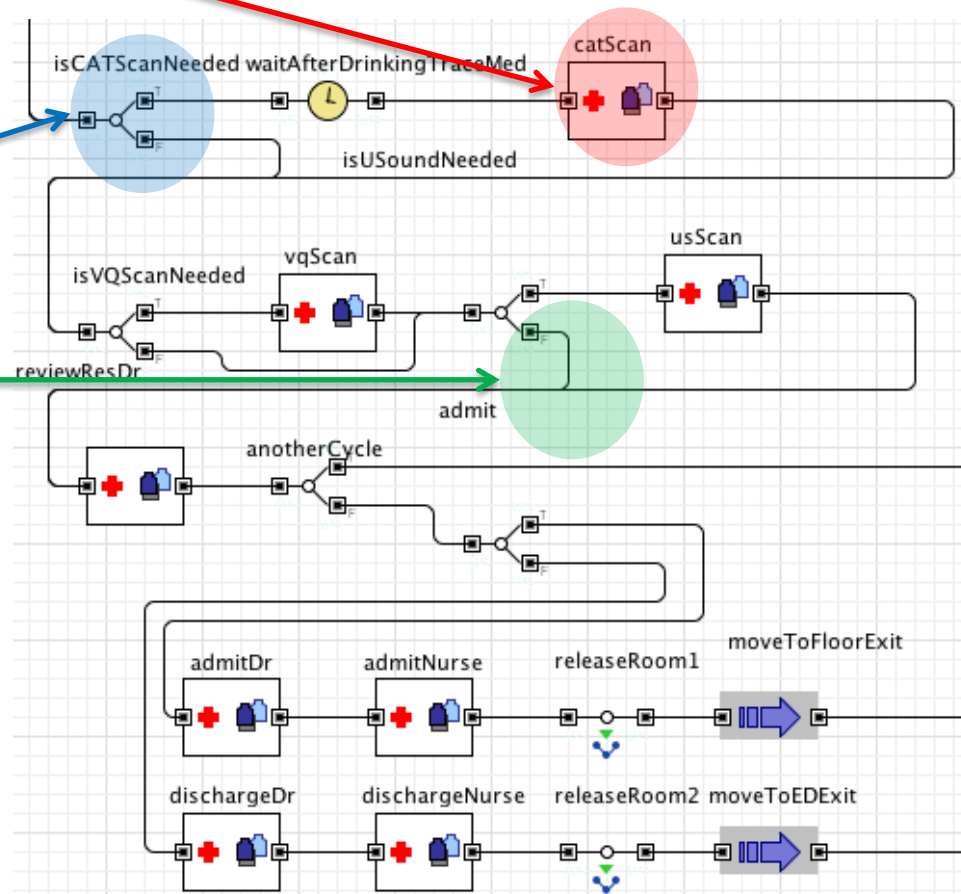
- **Finish**

- **Intermediate steps (icon indicates semantic kind)**



Flow Charts

- Flow charts can be **hierarchical**
- Frequently not linear e.g.
 - Branches
 - Joins



Palette Elements to Specify Flow Charts

The screenshot displays the AnyLogic Advanced [EDUCATIONAL USE ONLY] interface. The main workspace shows a flow chart with elements like 'release', 'moveToExit', 'networkExit', and 'sink'. A 'network' element is expanded to show 'doctor', 'procRoom', and 'scope' components. A red line highlights a path in the diagram. The left sidebar contains a project tree with 'MainPhase1' and 'Ports' expanded. The bottom panel shows the 'Properties' window for the 'doctor' element, which is a 'NetworkResourcePool'.

doctor - NetworkResourcePool

General
Name: Show Name Ignore Public Show At Runtime

Parameters
Type: Generic parameters:

Statistics
Package:

Description

Resource type:

Capacity defined: Directly By home shape By table over time

Capacity*:

Speed:

[^]New resource unit:

[^]On new unit:

Palette Elements:

- Model
- Action
- Analysis
- Presentation
- Connectivity
- Enterpris...
- Source
- Sink
- Hold
- Delay
- Queue
- Match
- Select Output
- Split
- Combine
- Resource Pool
- Seize
- Release
- Service
- Enter
- Exit
- Clock
- Conveyor
- Batch
- Unbatch
- Dropoff
- Pickup
- Restricted Ar...
- Restricted Ar...
- Network
- Network Enter...
- More Libraries...

Major Operators of Interest

- Source
- Sink
- Network enter/exit
(enter into a particular network)
- Select output (based on predicate)
- Split
- Delay
- Resource-related
 - Network seize/release: Association between Entity & Resource
 - Network attach/detach: Physically link entity & attached resources, so travel with entity
 - Network move to: Move entity (& movable attached resources) to particular place
 - Network send to: Send mobile resources to a location

Source: Entity Origination

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram and a flowchart. The flowchart starts with a 'source' block, which is highlighted with a red box. The flowchart includes blocks for 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. The 'source' block is connected to the 'networkEnter' block. The 'network' block is also visible in the diagram.

On the left, the 'Project' window shows a tree view of the model structure, including 'MainPhase3', 'Ports', 'Embedded Objects', and 'Presentation'. The 'Presentation' window shows a list of objects, including 'image', 'networkGroup', 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', 'rectangle', 'rectangle1', 'rectangle3', 'rectangle4', 'polyline', 'polyline1', and 'polyline4'. The 'Problems' window is also visible.

At the bottom, the 'Properties' window for the 'source' block is open. The 'General' tab shows the following configuration:

- Name: source
- Show Name:
- Ignore:
- Public:
- Show At Runtime:
- Create Presentation:
- Type: Source<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Arrivals defined by: Rate Interarrival time Rate table Arrival table Manual (call inject() method)
- Arrival rate*: 0.05
- Entities per arrival: 1
- Limited number of arrivals:
- New entity: new Entity()
- On exit:

A red arrow points from the text 'Defines rules governing origination of Entities to enter into network' to the 'Arrival rate*' field in the Properties window.

Defines rules governing origination of Entities to enter into network

Major Operators of Interest

- Source
- Sink
- Network enter/exit
(enter into a particular network)
- Select output (based on predicate)
- Split
- Delay
- Resource-related
 - Network seize/release: Association between Entity & Resource
 - Network attach/detach: Physically link entity & attached resources, so travel with entity
 - Network move to: Move entity (& movable attached resources) to particular place
 - Network send to: Send mobile resources to a location

Sink: Entity Cessation

The screenshot displays the AnyLogic University software interface. The main workspace shows a simulation model of a hospital layout with a flowchart on the right. The flowchart consists of three parallel paths:

- Path 1: source → networkEnter → networkSeize → sendToStorage → sendToPatient
- Path 2: networkAttach → moveToProcRoom → procedure → networkDetach → returnScope
- Path 3: release → moveToExit → networkExit → sink

The 'sink' entity is highlighted in a blue box in the flowchart. Below the workspace, the Properties window for the 'sink' entity is open, showing the following configuration:

sink - Sink

General
Name: sink Show name Ignore Public Show at runtime [Create presentation](#)

Parameters
Type: Sink<T extends Entity> Entity class: Entity

Statistics

Description
On enter:

Package: com.xj.anylogic.libraries.enterprise

Replicated
Initial number of objects:

At the bottom of the interface, the status bar shows 'sink - Sink', 'Selection', and coordinates 'X=679, Y=300'.

Major Operators of Interest

- Source
- Sink
- Network enter/exit
(enter into a particular network)
- Select output (based on predicate)
- Split
- Delay
- Resource-related
 - Network seize/release: Association between Entity & Resource
 - Network attach/detach: Physically link entity & attached resources, so travel with entity
 - Network move to: Move entity (& movable attached resources) to particular place
 - Network send to: Send mobile resources to a location

Network Enter: Informing Newly Created Entities of the Available Resources

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram. The network diagram consists of several interconnected nodes and flows. The nodes are labeled: source, networkEnter, networkSeize, sendToStorage, sendToPatient, networkAttach, moveToProcRoom, procedure, networkDetach, returnScope, release, moveToExit, networkExit, and sink. The networkEnter node is highlighted with a blue box. The network diagram is connected to a floor plan visualization on the left.

The Properties window for the selected **networkEnter - NetworkEnter** entity is visible at the bottom. The properties are as follows:

- General:** Name: networkEnter, Show Name, Ignore, Public, Show At Runtime, Create Presentation
- Parameters:** Type: NetworkEnter<T extends Entity>, Generic parameters: Entity
- Statistics:**
- Description:**
- Network*:** network
- ^Entry node*:** waitingHall
- ^On enter:**
- ^Speed:** 10
- Replication:**

The left sidebar shows a project tree with the following structure:

- MainPhase3
 - Ports
 - Embedded Objects
 - Presentation
 - image
 - networkGroup
 - waitingHall
 - staffRoom
 - storageRoom
 - procRoom1
 - procRoom2
 - exit
 - rectangle
 - rectangle1
 - rectangle3
 - rectangle4
 - polyline
 - polyline1
 - polyline4

The right sidebar shows a Palette with various simulation elements:

- Model
- Action
- Analysis
- Presentation
- Connectivity
- Enterprise...
- Source
- Sink
- Hold
- Delay
- Queue
- Match
- Select Output
- Split
- Combine
- Resource Pool
- Seize
- Release
- Service
- Enter
- Exit
- Clock
- Conveyor
- Batch
- Unbatch
- Dropoff
- Pickup
- Restricted Are...
- Restricted Are...
- Network
- Network Enter
- More Libraries...

Network Exit

The screenshot displays the AnyLogic University software interface. The main workspace shows a network diagram with nodes and connections. A red line highlights a path from a building icon to a 'networkExit' node, which is highlighted with a blue box. Below the diagram, the 'networkExit - NetworkExit' properties panel is visible, showing the following details:

- Name:** networkExit
- Show name
- Ignore
- Public
- Show at runtime
-
- Type:** NetworkExit<T extends Entity>
- Entity class:** Entity
- On enter^D:** [Empty field]
- Package:** com.xj.anylogic.libraries.enterprise
- Replicated
- Initial number of objects:** [Slider]

The bottom status bar shows 'Selection' and coordinates 'X=574, Y=304'.

Major Operators of Interest

- Source
- Sink
- Network enter/exit (enter into a particular network)
- **Select output (based on predicate)**
- Split
- Delay
- Resource-related
 - Network seize/release: Association between Entity & Resource
 - Network attach/detach: Physically link entity & attached resources, so travel with entity
 - Network move to: Move entity (& movable attached resources) to particular place
 - Network send to: Send mobile resources to a location

Determining factor can either be deterministic (e.g. based on condition) or stochastic (based on probability)
Select Output: Which Path Does Entity Take?

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model for an Emergency Department. A flowchart starts with 'Incoming patients' entering an 'EC Scenario ver. 1.2' block. From there, the flow goes to 'assAndPrep_PA_Tech', then to a 'SelectOutput' block named 'isXRayNeeded'. This block is highlighted with a blue circle and a blue arrow pointing to its properties window. The 'isXRayNeeded' block has two outgoing paths: one leading to 'seizeXRayRoom' and another leading to 'waitForXRayResults'. The 'seizeXRayRoom' path continues through 'moveToXRay', 'process', 'releaseXRayRoom', and 'moveFromXRay'. The 'waitForXRayResults' path leads to 'reviewTestResults_Doctor'. The 'reviewTestResults_Doctor' block has an output labeled 'Serviced patients' that loops back to the 'EC Scenario' block. The 'Properties' window for 'isXRayNeeded - SelectOutput' is open at the bottom, showing the following settings:

- Name: isXRayNeeded
- Show Name:
- Ignore:
- Public:
- Show At Runtime:
- Create Presentation:
- Type: SelectOutput<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Select True output: If condition is true With specified probability [0..1]
- Probability*: $xrayPatients / 100.0$
- On enter:
- On exit (true):
- On exit (false):

Major Operators of Interest

- Source
- Sink
- Network enter/exit (enter into a particular network)
- Select output (based on predicate)
- Split
- Delay
- Resource-related
 - Network seize/release: Association between Entity & Resource
 - Network attach/detach: Physically link entity & attached resources, so travel with entity
 - Network move to: Move entity (& movable attached resources) to particular place
 - Network send to: Send mobile resources to a location

Network Delay

The screenshot displays the AnyLogic Advanced software interface for an educational use only. The main workspace shows a network simulation model. On the left, a tree view lists the model's structure, including 'Ophthalmology Department', 'MainPhase1', 'Ports', 'Embedded Objects', and various objects like 'source', 'sink', 'networkEnter', 'networkExit', 'network', 'doctor', 'procRoom', 'scope', 'procedure', 'moveToExit', and 'moveToProcRoom'. The main workspace contains a floor plan of a department and a flow diagram. The flow diagram starts with a 'source' object, followed by a 'networkEnter' object, then a 'moveToProcRoom' object, a 'procedure' object (highlighted with a purple box), a 'moveToExit' object, a 'networkExit' object, and finally a 'sink' object. The 'procedure' object is configured with a delay.

procedure - Delay

General

Name: Show Name Ignore Public Show At Runtime

Parameters

Type: Generic parameters:

Package:

Delay time is Specified explicitly Path length / speed

[^]Delay time*

Capacity*

Maximum capacity

[^]On enter

[^]On exit

Selection

Recall: Resources

- Frequently resources are required to initiate a particular phase of processing
 - A doctor (resource) to administer surgery to a patient (entity)
 - A piece of diagnostic equipment (resource) to image a patient (entity)
 - An EKG to (resource) to record from a patient (entity)
 - A gurney or bed (resource) for a patient (entity)
- Distinctions amongst these resources
 - Portable vs. fixed
 - Mobile (with agency)
- To capture these dependencies, a network is often associated with multiple types of resources

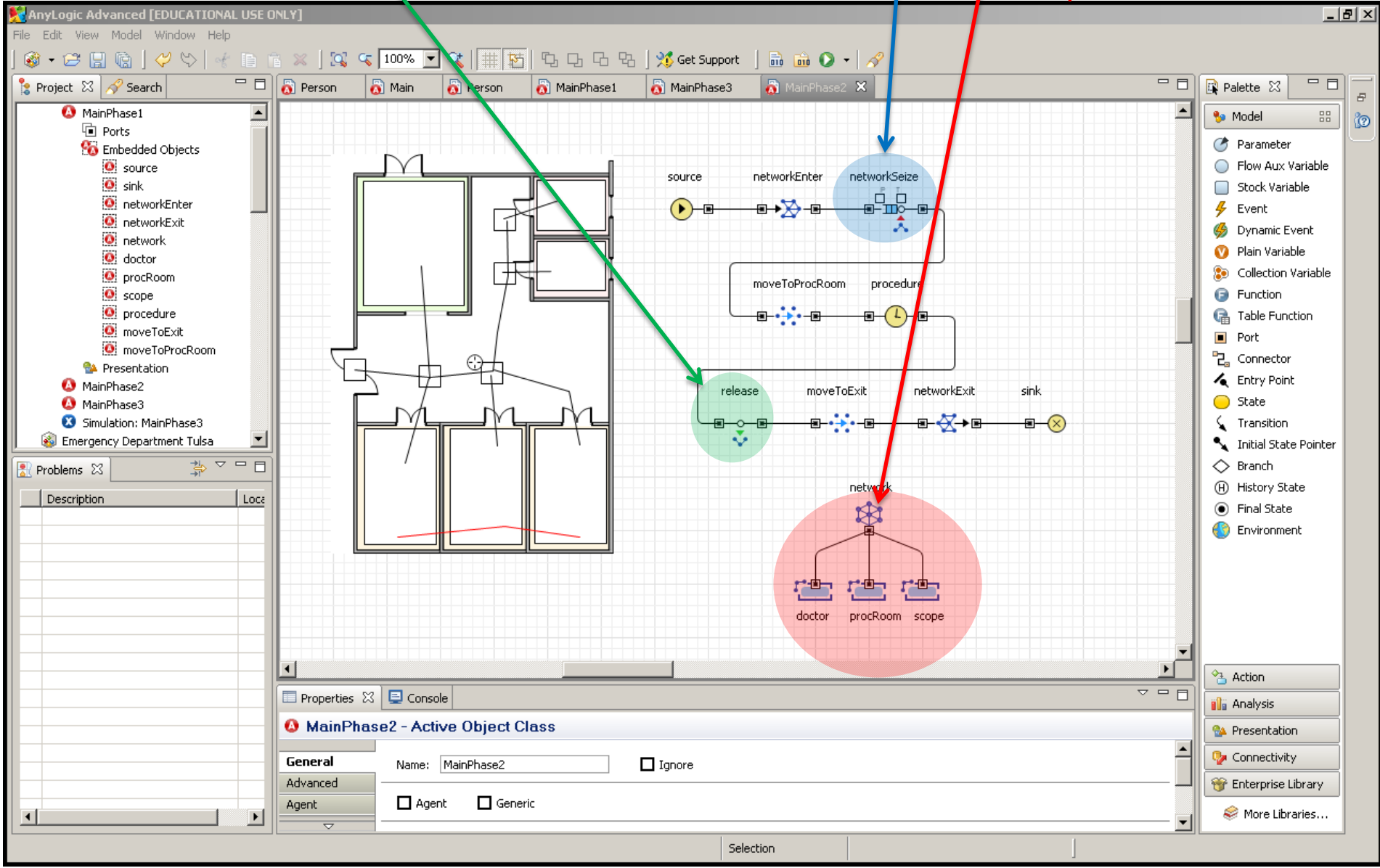
Resources 2

- When an agent cannot obtain (“seize”) a resource, they “enqueue” and wait for that resource to be released by another entity
 - These resources live in “pools” of interchangeable “resource units”
 - A “seized” resource comes from the pool
 - A “released” resource returns to the pool
 - If wish to be able to choose particular resources from a pool, create in *different pools*, and select desired pool

“Seizes” (seeks to achieve exclusive association with) a resource (otherwise waits)
“Releases” association with a resource, so others can be associated with/use it

Resources

Types of resources associated with the network, each in a resource pool



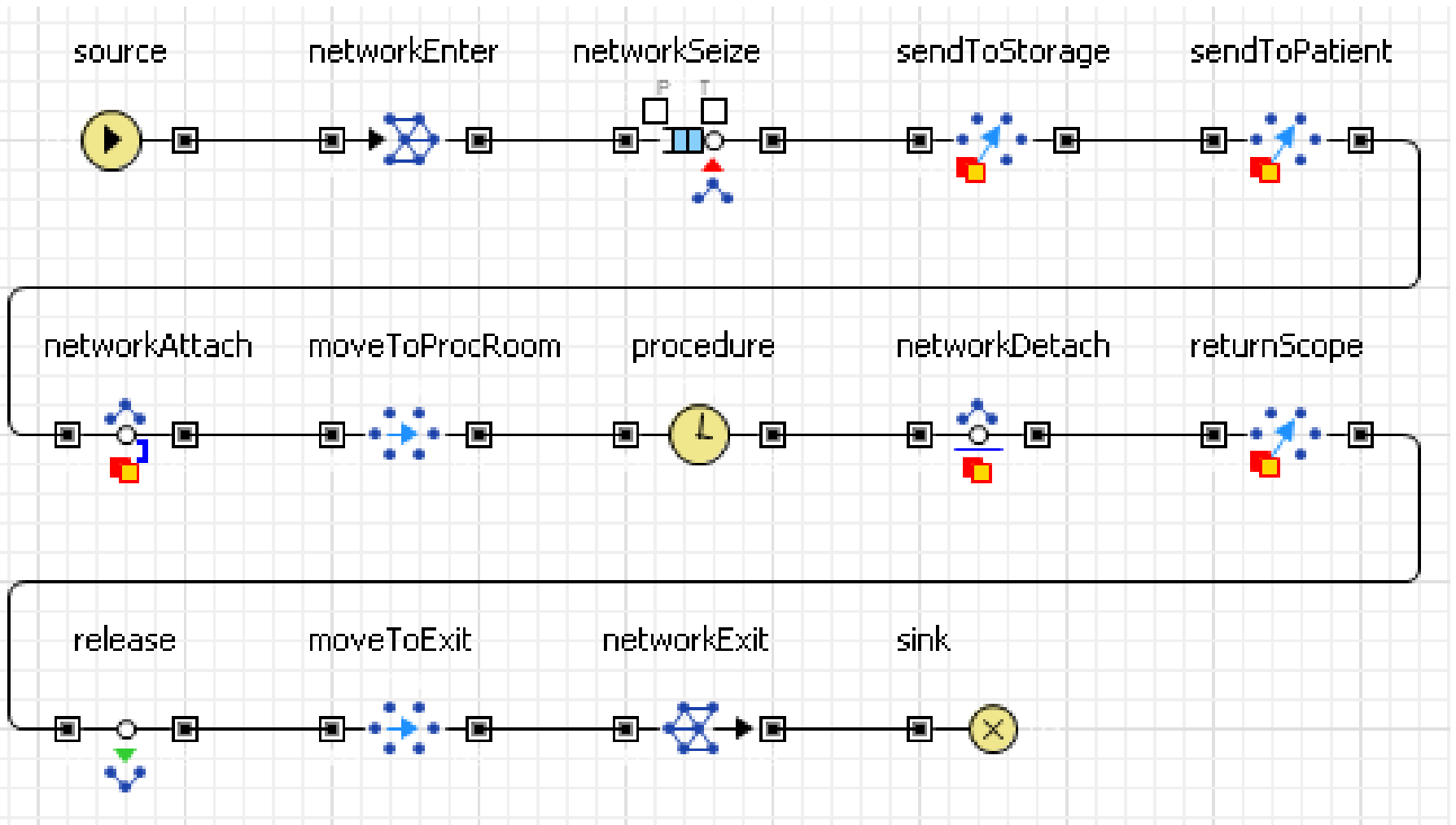
Main Flow Operators Associated with Resources

- All resources
 - Network Seize
 - Network Release
- All non-static (portable&mobile resources)
 - Network Attach (NetworkAttach)/Detach (NetworkAttach)
- Moving entity – along with attached portable & mobile resources – to a location
 - NetworkMoveTo
- Moving mobile resources to a location
 - Network SendTo (NetworkSendTo)

A Key Distinction

- Seizing/Releasing a resource: Is this resource reserved by (uniquely associated with) the entity?
- Attaching/Detaching a resource: Will this resource spatially follow the agent?

Another Flow Chart



Securing Association with 1 or More Resources: Network Seize

Resource pools with whose resources Entity is seeking association

This seizes one resource unit from each pool

One resource may be seized while waiting for ("blocking for") the other.

networkSeize - NetworkSeize

General Name: networkSeize Show Name Ignore Public Show At Runtime

Parameters Type: NetworkSeize<T extends Entity> Generic parameters: Entity

Description Package: com.xj.anylogic.libraries.enterprise

^List of resources { pool, ... } **{procRoom, doctor, scope}**

^On enter

^On exit

Queue capacity

Maximum queue capacity

Enable exit on timeout

Enable preemption

Network Release: Dissociating Entity & Resource

The screenshot displays the AnyLogic software interface. On the left, a project tree shows a hierarchy of models including 'Main', 'Simulation: Main', 'Schelling Segregation', 'ABMModelWithBirthDeath', 'SIR Agent Based Calibration', and 'OphthalmologyDepartmentAnylogic'. The main workspace is divided into two parts: a network diagram on the left and a flowchart on the right. The network diagram shows a building layout with rooms and doors, connected to a flowchart. The flowchart consists of three horizontal paths of nodes: 1) 'source' (yellow circle) -> 'networkEnter' (blue square) -> 'networkSeize' (blue square with red triangle) -> 'sendToStorage' (red square) -> 'sendToPatient' (red square). 2) 'networkAttach' (blue square) -> 'moveToProcRoom' (blue square) -> 'procedure' (yellow circle) -> 'networkDetach' (blue square) -> 'returnScope' (red square). 3) 'release' (blue square) -> 'moveToExit' (blue square) -> 'networkExit' (blue square) -> 'sink' (yellow circle). A purple line connects the 'release' node to the 'networkExit' node. Below the workspace, the 'Properties' panel is open for a 'release' entity. The 'General' tab shows: Name: 'release', 'Show name' checked, 'Ignore' unchecked, 'Public' unchecked, 'Show at runtime' checked, and a 'Create presentation' button. The 'Parameters' tab shows: Type: 'NetworkRelease<T extends Entity>', Entity class: 'Entity'. The 'Release' section has 'Specified resources' selected. The 'Moving resources' section has 'Return to home location' selected. There are input fields for 'On enter' and 'On exit'. The 'Package' is set to 'com.xj.anylogic.libraries.enterprise'. A 'Replicated' checkbox is at the bottom.

AnyLogic University [EDUCATIONAL USE ONLY]
File Edit View Draw Model Tools Help

Projects
MainPhase1 MainPhase2 MainPhase3

release - NetworkRelease

General
Name: Show name Ignore Public Show at runtime

Parameters
Type: NetworkRelease<T extends Entity> Entity class:

Release Specified resources All seized resources

Moving resources Return to home location Stay where they are

On enter
On exit

Package:

Replicated

Network Send To: Moving a (Seized) Resource to a Resource, Entity, or Place

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram and a flowchart. The flowchart includes nodes for 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. The 'sendToStorage' node is highlighted with a blue box. The Properties window for 'sendToStorage - NetworkSendTo' is open, showing the following configuration:

- Name: sendToStorage
- Type: NetworkSendTo<T extends Entity>
- Package: com.xj.anylogic.libraries.enterprise
- Resources to send*: { doctor }
- Destination is*: Specified node Entity Seized resource unit Home of seized resource unit
- Resource*: scope

Annotations in the image include:

- A red arrow pointing from the text 'Here, sending an already seized resource to another resource' to the 'Resources to send*' field.
- A blue arrow pointing from the text 'Here, sending an already seized resource to another resource' to the 'Resource*' field.

Example of Simultaneously Moving Multiple Resources Together via SendTo

AnyLogic Advanced [EDUCATIONAL USE ONLY]

File Edit View Model Window Help

100%

Get Support

Project Search

Ophthalmology Department

- MainPhase1
 - Ports
 - Embedded Objects
 - source
 - sink
 - networkEnter
 - networkExit
 - network
 - doctor
 - procRoom
 - scope
 - procedure
 - moveToExit
 - moveToProcRoom
 - Presentation
 - MainPhase2
 - MainPhase3
 - Simulation: MainPhase3

Person Main MainPhase1 MainPhase3 MainPhase2 ECProcess

source networkEnter networkSeize sendToStorage sendToPatient

networkAttach moveToProcRoom procedure networkDetach returnScope

release moveToExit networkExit sink

Palette

- Model
- Action
- Analysis
- Presentation
- Connectivity
- Enterprise...

Source

Sink

Hold

Delay

Queue

Match

Select Output

Split

Combine

Resource Pool

Seize

Release

Service

Enter

Exit

Clock

Conveyor

Batch

Unbatch

Dropoff

Pickup

Restricted Are...

Restricted Are...

Network

Manual Entry

More Libraries...

Problems

Description

Loca

Properties Console

sendToPatient - NetworkSendTo

General Name: sendToPatient Show Name Ignore Public Show At Runtime Create Presentation

Parameters Type: NetworkSendTo<T extends Entity> Generic parameters: Entity

Statistics Package: com.xj.anylogic.libraries.enterprise

Description

Resources to send: {doctor, scope}

Destination is* Specified node Entity Seized resource unit Home of seized resource unit

On enter

On exit

Replication:

Here, sending 2 already Seized resource s to the entity (the patient)

Network Attach: Associating Entity with Specified Seized Resources, or those Nearby (So move together henceforth)

Project Tree:

- Ophthalmology Department
 - MainPhase1
 - Ports
 - Embedded Objects
 - source
 - sink
 - networkEnter
 - networkExit
 - network
 - doctor
 - procRoom
 - scope
 - procedure
 - moveToExit
 - moveToProcRoom
 - Presentation
 - MainPhase2
 - MainPhase3
 - Simulation: MainPhase3

Flowchart:

```
graph LR; source((source)) --> networkEnter[networkEnter]; networkEnter --> networkSeize[networkSeize]; networkSeize --> sendToStorage[sendToStorage]; sendToStorage --> sendToPatient[sendToPatient]; sendToPatient --> networkAttach[networkAttach]; networkAttach --> moveToProcRoom[moveToProcRoom]; moveToProcRoom --> procedure((procedure)); procedure --> networkDetach[networkDetach]; networkDetach --> returnScope[returnScope]; returnScope --> release[release]; release --> moveToExit[moveToExit]; moveToExit --> networkExit[networkExit]; networkExit --> sink((sink));
```

Properties Panel: networkAttach - NetworkAttach

General

Name: networkAttach Show Name Ignore Public Show At Runtime

Parameters

Type: NetworkAttach<T extends Entity> Generic parameters: Entity

Package: com.xj.anylogic.libraries.enterprise

Attach* All seized non-static resources at entity location Specified resources

On enter: _____

On exit: _____

Replication: _____

Palette:

- Model
- Action
- Analysis
- Presentation
- Connectivity
- Enterprise...
- Source
- Sink
- Hold
- Delay
- Queue
- Match
- Select Output
- Split
- Combine
- Resource Pool
- Seize
- Release
- Service
- Enter
- Exit
- Clock
- Conveyor
- Batch
- Unbatch
- Dropoff
- Pickup
- Restricted Are...
- Restricted Are...
- Network
- Network Enter
- More Libraries...

Note that this attachment Occurs after the Resources are in proximity with (have reached)

Network Move To: Moving an *Entity* to a Resource (or Node)

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows the hierarchy: Ophthalmology Department* > MainPhase1 > Embedded Objects > source, sink, networkEnter, networkExit, network, doctor, procRoom, scope, procedure, moveToExit, moveToProcRoom. The main workspace shows a network diagram with nodes and transitions. A transition labeled 'moveToProcRoom' is highlighted with a blue box. Below the diagram, the 'moveToProcRoom - NetworkMoveTo' configuration window is open. The 'Destination is*' section has 'Seized resource unit' selected, and the 'Resource*' field contains 'procRoom'. A red arrow points from the text 'Resource to which agent should move' to this field.

Resource to which agent should move (here, already seized unit from this Resource pool)

NB: Because resources are attached (seized), this MoveTo will Move Entity & but also bring moving & portable resources along (doctor & scope)

Network Detach

So **entity** can be physically Separated from **resources** (while remaining associated w/them)

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model of an ophthalmology department, including a network diagram with nodes like 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. The 'networkDetach' component is highlighted with a blue box.

The left sidebar shows the project structure, including 'Ophthalmology Department', 'MainPhase1', 'Ports', 'Embedded Objects', and 'Presentation'. The bottom-left pane shows the 'Problems' list.

The bottom-right pane shows the 'Properties' window for the 'networkDetach' component. The 'General' tab is active, showing the following details:

- Name: networkDetach
- Parameters: Show Name, Ignore, Public, Show At Runtime, Create Presentation
- Type: NetworkDetach<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Detach*: All attached resources Specified resources
- On enter: [Empty text field]
- On exit: [Empty text field]
- Replication: [Empty text field]

The right sidebar shows the 'Palette' with various simulation components like Model, Action, Analysis, Presentation, Connectivity, and Enterprise... The 'Enterprise...' palette includes components like Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, and Network.

Releasing Associated Resources

Resource pools whose Resource Units one is releasing

Moving resource (doctor) returns to home location after release of association with entity

release - NetworkRelease

General
Name: Show Name Ignore Public Show At Runtime

Parameters
Type: Generic parameters:

Description
Package:

Release: Specified resources All seized resources

Moving resources: Return to home location Stay where they are

On enter:

On exit:

Replication:

Visual Depiction Accompanying Logical Flow

- Entities are associated with icons
- Resources are associated with
 - Locations
 - Icons
- Movement networks are associated with routing paths
 - Often want to move resources or icons among different visual locations
 - Specific points (e.g. a storage closet for mobile resources)
 - Points associated with fixed resources (e.g. a MRI scanner)

Association of Network with Paths

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows a hierarchy: Ophthalmology Department* > MainPhase1 > Ports > Embedded Objects > sink, networkEnter, networkExit, network, doctor, procRoom, scope, procedure, moveToExit, moveToProcRoom. The main workspace shows a network diagram with nodes and connections. A red arrow points from the text 'Group of visual objects associated with network' to the 'networkGroup' property in the 'network - Network' properties window. The properties window shows: Name: network, Type: Network, Package: com.xj.anylogic.libraries.enterprise, Group of network shapes*: networkGroup, Hide network shapes: checked, When item is at a node: Draw at random position within node, Enable priorities: unchecked.

Group of visual objects associated with network

Associated Presentation “Group”

The network will “know” about these
(e.g. location & interconnectivity for routing)

The screenshot displays the AnyLogic Advanced software interface. The main workspace is divided into several panes:

- Project Explorer (Left):** Shows a tree view of the model structure. Under "Presentation", there is a "networkGroup" folder containing various elements like "waitingHall", "staffRoom", "storageRoom", "procRoom1", "procRoom2", "exit", and several "rectangle" and "polyline" objects.
- Main Canvas (Center):** Displays a simulation model. On the left, there is a floor plan diagram with three rooms. On the right, there is a network diagram with nodes labeled "networkAttach", "moveToProcRoom", "procedure", "networkDetach", "returnScope", "release", "moveToExit", "networkExit", and "sink". Below the network diagram, there are three nodes labeled "doctor", "procRoom", and "scope".
- Properties Panel (Bottom):** Shows the properties for the selected "networkGroup". The "General" tab is active, displaying the name "networkGroup" and several checkboxes: "Show Name" (unchecked), "Ignore" (unchecked), "Public" (checked), and "Icon" (unchecked).
- Palette (Right):** A vertical toolbar containing various simulation components such as Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, Network, and Network Entry.

Presentation of Entity

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation model with a network diagram. A 'source' entity is highlighted with a blue box. The Properties window for the 'source' entity is open, showing the following configuration:

- Name: source
- Type: Source<T extends Entity>
- Generic parameters: Entity
- Package: com.xj.anylogic.libraries.enterprise
- Arrivals defined by: Rate
- Arrival rate*: 0.05
- Entities per arrival: 1
- Limited number of arrivals:
- New entity: new Entity()
- On exit:
- Entity animation shape*: shapePatient
- Unique shape for each entity:
- Enable rotation:

A red arrow points from the text 'Appearance of the entity when moving through the network' to the 'shapePatient' property in the Properties window.

Appearance of the entity when moving through the network

Presentation Properties of a Resource

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows a resource group named 'network' containing several resource units, with 'staffRoom' highlighted in green. The main workspace shows a 3D architectural model of a hospital and a Petri net diagram. The Petri net diagram includes transitions like 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. Resource units 'doctor', 'procRoom', and 'scope' are shown in blue boxes. A red arrow points from the text 'Appearance Of the resource units in this resource in Idle & Busy States' to the 'shapeDoctor' property in the 'doctor - NetworkResourcePool' properties window. A blue arrow points from the text 'Home position of resource in presentation group associated with network' to the 'staffRoom' property in the same window.

Appearance Of the resource units in this resource in Idle & Busy States

Home position of resource in presentation group associated with network

doctor - NetworkResourcePool

Property	Value
On seize	
On release	
Idle unit animation shape*	shapeDoctor
Busy unit animation shape*	shapeDoctor
Unique shape for each unit	<input type="checkbox"/>
Enable rotation	<input type="checkbox"/>
Home defined by	Single node
Home node*	staffRoom
Enable statistics	<input type="checkbox"/>

Entering the Network: Where & with What (Logical & Presentation) Network

The screenshot displays the AnyLogic Advanced interface. On the left, a project tree shows a network structure with various rooms and entities. The main workspace contains a network diagram with nodes like 'source', 'networkEnter', 'networkSeize', 'sendToStorage', 'sendToPatient', 'networkAttach', 'moveToProcRoom', 'procedure', 'networkDetach', 'returnScope', 'release', 'moveToExit', 'networkExit', and 'sink'. A red arrow points from the 'Speed' property in the configuration window to the 'networkEnter' node in the diagram.

networkEnter - NetworkEnter

General

Name: networkEnter Show Name Ignore Public Show At Runtime

Parameters

Type: NetworkEnter <T extends Entity > Generic parameters: Entity

Package: com.xj.anylogic.libraries.enterprise

Network*: network

^Entry node*: waitingHall

^On enter:

^Speed: 10

Replication:

Speed to use as Entities move around

Movement Network: Defined by Polygons & Rectangles

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a floor plan of an Emergency Department with a blue-shaded area representing a movement network. A logical network diagram is overlaid on the floor plan, showing a sequence of actions: source, networkEnter, networkSeize, sendToStorage, sendToPatient, networkAttach, moveToProcRoom, procedure, networkDetach, returnScope, release, moveToExit, networkExit, and sink. The project tree on the left lists various elements, including a networkGroup and several polylines. The palette on the right contains various modeling elements like Source, Sink, Hold, Delay, Queue, Match, Select Output, Split, Combine, Resource Pool, Seize, Release, Service, Enter, Exit, Clock, Conveyor, Batch, Unbatch, Dropoff, Pickup, Restricted Area, and Network.

Recall: This is the visual (presentation) network associated with the logical network

These "polylines" and rectangles are the elements over which the Entities & Resources move

Recall: The Location of Room Resource Pool is given as being “Path across nodes” defined by the Polyline

The screenshot shows the AnyLogic Advanced software interface. The main workspace displays a simulation model of an emergency department. The top part of the workspace shows a floor plan with a red line indicating a path across nodes. Below the floor plan is a network diagram with nodes labeled 'doctor', 'procRoom', and 'scope'. The 'procRoom' node is highlighted with a blue box. The Properties window at the bottom shows the configuration for 'procRoom - NetworkResourcePool'. The 'Home path*' field is set to 'roomsLocation', and a dropdown menu shows 'Path across nodes'.

procRoom - NetworkResourcePool

Property	Value
General	
Parameters	
Statistics	
Description	
On seize	
On release	
Idle unit animation shape	
Busy unit animation shape	
Unique shape for each unit	<input type="checkbox"/>
Enable rotation	<input type="checkbox"/>
Home defined by*	Path across nodes
Home path*	roomsLocation
Enable statistics	<input type="checkbox"/>

Polyline Describes the Location of the Procedure Rooms

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a floor plan of a hospital emergency department with a red-outlined polygon highlighting a specific area. A red arrow points from a text box to the vertices of this polygon. The text box contains the following text:

The rectangles touched by this poly line vertices are the room locations

The properties panel at the bottom shows the configuration for the 'roomsLocation - Polyline' object:

- Name: roomsLocation
- Show Name:
- Ignore:
- Public:
- Icon:
- Fill Color: No Fill
- Line Color: red
- Line Width: 1 pt
- Line Style: Solid
- Polyline options: Close Polyline

Moving Entity to an Explicit Visual Point

The screenshot displays the AnyLogic University interface. On the left, a project tree for 'MainPhase3' lists various objects, with 'exit' highlighted in blue. A blue arrow points from this 'exit' object to the 'moveToExit' node in the network diagram. The network diagram shows a flow starting from a 'release' node, passing through a 'moveToExit' node (highlighted with a blue box), then through 'networkExit' and 'sink' nodes. Below the network diagram, a 'network' block is shown with 'doctor', 'procRoom', and 'scope' nodes. The bottom panel shows the 'Properties' window for the 'moveToExit' node, where the 'Destination is' is set to 'Specified node' and the 'Node' is set to 'exit'.

“exit” is presentation node defined here

moveToExit - NetworkMoveTo

General
Name: moveToExit Show name Ignore Public Show at runtime

Type: NetworkMoveTo<T extends Entity> Entity class: Entity

Destination is Specified node Seized resource unit

Node^D **exit**

On enter^D

On exit^D

moveToExit - NetworkMoveTo Selection X=483, Y=298

Subclassing: A Valuable Tool

- So as to customized the desired system behavior, it can be useful to customize entities & resources (resource units)
 - To e.g. carry around additional information (e.g. associated external agent in agent-based model, history information, etc.)
 - Particular specialized network types
- Because the original entities & resource units are classes, this can be accomplished via subclassing (subclass Entity & ResourceUnit)
- If do this, parameterize generics by subclass type S (e.g. NetworkResourcePool<S>)
- We will be discussing subclassing in an upcoming Java tutorial

Understanding the Model: MainPhase1

The screenshot displays the AnyLogic software interface. The main workspace shows the 'Ophthalmology Department' experiment setup page, which includes a title, a subtitle 'Experiment setup page', a button 'Run the model and switch to Main view', and a description of the 'Enterprise Library Tutorial model'. The left sidebar contains a project tree with various elements like 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', 'rectangle', 'polyline', 'roomsLocation', 'shapeDoctor', 'shapePatient', and 'shapeScope'. The bottom panel shows the 'Simulation - Simulation Experiment' configuration window, where the 'Main active object class (root)' is set to 'MainPhase1'. A red circle highlights the dropdown menu for this setting, and a red arrow points from the text 'Setting simulation to Run MainPhase1 Main class' to the 'MainPhase1' option in the dropdown.

Ophthalmology Department
Experiment setup page

Run the model and switch to Main view

Enterprise Library Tutorial model
Plays the role of a reference model for the tutorial (available from Help|Help Contents -> Enterprise Library Tutorial section).

The simplest network-based model of a typical ophthalmology unit. Patients arrive to the department to undergo the ophthalmoscopy procedure. They are held in the waiting room and wait for ophthalmologists to come and make an examination. The procedure is held in the procedure room. When a doctor arrives, patient moves to empty procedure room escorted by the doctor. The procedure is performed using an ophthalmoscope. Ophthalmoscopes are stored in the storage room and are taken by doctors just before the procedure begins. Following the procedure, the doctor transports the ophthalmoscope back to the storage room and returns to the staffroom, and the patient leaves the ophthalmology department.

Simulation - Simulation Experiment

General Name: Simulation Main active object class (root): MainPhase1 Ignore

Random number generation:

- Random seed (unique simulation runs)
- Fixed seed (reproducible simulation runs) Seed value: 1
- Custom generator (subclass of Random): new Random()

Setting simulation to Run MainPhase1 Main class

No Resource Constraints

The screenshot displays the AnyLogic University software interface. The main workspace is divided into two parts: a floor plan on the left and a network flow diagram on the right. The floor plan shows a hospital department layout with rooms like 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', and 'exit'. The network flow diagram illustrates the movement of agents through the system, starting from a 'source' and passing through 'networkEnter', 'moveToProcRoom', 'procedure', 'moveToExit', 'networkExit', and finally reaching a 'sink'. A 'network' component is also shown at the bottom, connected to 'doctor', 'procRoom', and 'room' agents.

The interface includes a menu bar (File, Edit, View, Draw, Model, Tools, Help), a toolbar, and a 'Projects' panel on the left. The 'Projects' panel shows a tree view of the model structure, including 'Person', 'Calibration: Main', 'MonteCarlo2DHistogram: Mair', 'OphthalmologyDepartmentAnylogic', 'MainPhase1', 'MainPhase2', 'MainPhase3', 'Embedded Objects', 'Connectors', 'Presentation', 'image', 'networkGroup', 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', 'rectangle', 'rectangle1', 'rectangle3', 'rectangle4', and 'polyline'.

The 'Problems' panel at the bottom left shows 'No problems'. The 'Properties' panel at the bottom right is titled 'MainPhase1 - Active Object Class' and contains the following information:

- General**: Name: MainPhase1, Ignore
- Advanced**
- Agent**: Agent, Generic
- Preview**
- Description**: Startup code: [text area], Destroy code: [text area]

The status bar at the bottom indicates 'Selection' and 'X=377, Y=299'.

Entering the Network

The screenshot displays the AnyLogic software interface. On the left, a project tree shows a hierarchy of objects, with 'networkGroup' containing 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', 'rectangle', 'rectangle1', 'rectangle3', 'rectangle4', and 'polyline'. A blue arrow points from 'waitingHall' in the tree to the 'networkEnter' block in the main diagram. The main diagram shows a network layout with a 'source' block connected to a 'networkEnter' block, which is connected to a 'moveToProcRoom' block, then a 'procedure' block, then a 'moveToExit' block, then a 'networkExit' block, and finally a 'sink' block. A green arrow points from the 'Speed' field in the 'networkEnter' properties to the 'Speed' field in the 'network' properties. The 'network' properties show 'Speed' set to '10'. A green text annotation 'Speed specified here' is placed near the 'networkExit' block. A red text annotation 'Entities enter At waiting hall' is placed near the 'waitingHall' block in the project tree.

Entities enter
At waiting hall

Speed specified here

networkEnter - NetworkEnter

Property	Value
Name	networkEnter
Type	NetworkEnter <T extends Entity>
Entity class	Entity
Network	network
Entry node	waitingHall
Speed	10

Moving to a Fixed Node

The screenshot displays the AnyLogic University interface. On the left, a project tree shows a hierarchy of models, with 'procRoom1' selected under 'MainPhase1'. The main workspace is divided into two panels: a floor plan on the left and a network diagram on the right. The network diagram shows a flow from a 'source' node through 'networkEnter', a 'moveToProcRoom' node (highlighted with a purple box), a 'procedure' node, 'moveToExit', 'networkExit', and finally to a 'sink' node. A 'network' node is also visible at the bottom. The bottom panel shows the 'Properties' window for the 'moveToProcRoom' node, where the 'Destination is' is set to 'Specified node' and the 'Node' is set to 'procRoom1'. A red arrow points from the text 'Entities move to specific room' to the 'procRoom1' node in the network diagram.

Entities move to specific room

moveToProcRoom - NetworkMoveTo

Selection X=480, Y=199

Uniformly Distributed Delay

The screenshot displays the AnyLogic University interface for an educational simulation. The main workspace shows a floor plan of a hospital with a network flow diagram overlaid. The flow starts at a 'source' (yellow circle), moves through 'networkEnter' (blue diamond), then to 'moveToProcRoom' (blue star), and enters a 'procedure' block (blue rectangle) which contains a delay element (yellow circle with a clock). From the procedure, the flow goes to 'moveToExit' (blue star), then through 'networkExit' (blue diamond), and finally to a 'sink' (yellow circle with an 'X'). A 'network' node (blue diamond) is also shown at the bottom of the floor plan.

The left sidebar shows a project tree with the following structure:

- Person
- Calibration: Main
- MonteCarlo2DHistogram: Main
- OphthalmologyDepartmentAnylogic
 - MainPhase1
 - MainPhase2
 - MainPhase3
 - Embedded Objects
 - Connectors
 - Presentation
 - image
 - networkGroup
 - waitingHall
 - staffRoom
 - storageRoom
 - procRoom1
 - procRoom2
 - exit
 - rectangle
 - rectangle1
 - rectangle3
 - rectangle4
 - polyline

The bottom panel shows the configuration for the 'procedure - Delay' element:

- General**: Name: procedure, Show name, Ignore, Public, Show at runtime,
- Parameters**: Type: Delay<T extends Entity>, Entity class: Entity
- Statistics**: Delay time is: Specified explicitly, Path length / speed
- Description**: Delay time^D: uniform(10), Capacity: 5, Maximum capacity:

The status bar at the bottom indicates 'procedure - Delay', 'Selection', and coordinates 'X=558, Y=196'.

To see Capacity Limitation, Increase Speed

The screenshot displays the AnyLogic University software interface. The main workspace is divided into two parts: a floor plan on the left and a network flow diagram on the right. The floor plan shows a hospital layout with rooms and corridors. The network flow diagram illustrates the flow of entities through the system, starting from a 'source' node, passing through a 'networkEnter' node, then through a 'moveToProcRoom' node, a 'procedure' node (represented by a clock icon), a 'moveToExit' node, a 'networkExit' node, and finally reaching a 'sink' node. A 'network' node is also shown at the bottom of the diagram.

The 'Projects' panel on the left shows a tree view of the simulation model, including 'Person', 'Calibration: Main', 'MonteCarlo2DHistogram: Mair', 'OphthalmologyDepartmentAnylogic', 'MainPhase1', 'MainPhase2', 'MainPhase3', 'Embedded Objects', 'Connectors', 'Presentation', 'image', 'networkGroup', 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', 'rectangle', 'rectangle1', 'rectangle3', 'rectangle4', and 'polyline'.

The 'Properties' panel at the bottom shows the configuration for the 'networkEnter' entity. The 'General' tab is selected, showing the following settings:

- Name: networkEnter
- Show name:
- Ignore:
- Public:
- Show at runtime:
- Create presentation:

The 'Parameters' tab is also visible, showing the following settings:

- Type: NetworkEnter <T extends Entity>
- Entity class: Entity
- Network: network
- Entry node: waitingHall
- On enter:
- Speed: 100

Limited Capacity

Ophthalmology Department : Simulation - AnyLogic University [EDUCATIONAL USE ONLY]

root:MainPhase1

AnyLogic

source networkEnter

301,081 301,081 301,081

moveToProcRoom procedure

0 4

Exception during discrete event execution

RuntimeException: root.moveToProcRoom:
out: An entity was not able to leave this port
Consider increasing capacities and/or throughputs of the subsequent object(s)
Please see AnyLogic Console View for more details.

OK

sink

301,055

network

0

5/15 doctor 3/3 procRoom 5/5 scope

Run: 0 Error Time: 6026451.62 Simulation: Stop time not set Memory: 23M of 59M 5.4 sec

Moving to Exit

The screenshot displays the AnyLogic University interface. On the left, the Project Explorer shows a tree structure with 'MainPhase1', 'MainPhase2', and 'MainPhase3' under 'Embedded Objects'. The 'exit' node is highlighted in blue. The main workspace shows a floor plan of a hospital with rooms and corridors. To the right, a network diagram illustrates the simulation logic. It starts with a 'source' node, followed by 'networkEnter', 'moveToProcRoom', 'procedure', 'moveToExit', 'networkExit', and 'sink'. The 'moveToExit' node is highlighted with a blue box. The Properties panel for 'moveToExit' is open, showing the 'Node' property set to 'exit'.

Entities move to specific Presentation location

moveToExit - NetworkMoveTo

General Name: moveToExit Show name Ignore Public Show at runtime

Parameters Type: NetworkMoveTo<T extends Entity> Entity class: Entity

Statistics

Description Destination is Specified node Seized resource unit

Node^D

On enter^D

On exit^D

Selection X=525, Y=302

Phase 2

The screenshot displays the AnyLogic University interface. On the left, a project tree shows a hierarchy of models, with 'MainPhase2' selected. The main workspace is divided into two panels: a floor plan on the left and a flowchart on the right. The floor plan shows a building layout with rooms and corridors, with a red line indicating a path. The flowchart illustrates the logic of the simulation, starting with a 'source' node, followed by 'networkEnter', 'networkSeize', 'moveToProcRoom', 'procedure', 'release', 'moveToExit', 'networkExit', and ending at a 'sink' node. A 'network' object is also shown at the bottom of the flowchart.

Projects

- Simulation: Main
 - Schelling Segregation
 - Main
 - Person
- Simulation: Main
 - Person
 - IAgentAction
 - IAgentSpecification
 - IMainAction
- Simulation: Main
 - SIR Agent Based Calibration
 - Main
 - Person
 - Calibration: Main
 - MonteCarlo2DHistogram: Main
- OphthalmologyDepartmentAnylogic
 - MainPhase1
 - MainPhase2**
 - MainPhase3
 - Embedded Objects
 - Connectors

Problems

No problems

Description	Locat...
-------------	----------

MainPhase2 - Active Object Class

General Name: Ignore

Advanced

Agent Agent Generic

Preview

Description Startup code:

Entering the Network: Same

The screenshot displays the AnyLogic University interface. On the left, a 'Projects' tree shows a 'networkGroup' containing 'waitingHall', 'staffRoom', and 'procRoom'. The main workspace shows a floor plan with a network diagram. A flowchart on the right details the logic: a 'source' leads to 'networkEnter', which connects to 'networkSeize', then 'moveToProcRoom', 'procedure', 'release', 'moveToExit', 'networkExit', and finally 'sink'. A 'network' object is also shown at the bottom.

The 'Properties' window for 'networkEnter - NetworkEnter' is open at the bottom. It shows the following configuration:

- Name: networkEnter
- Type: NetworkEnter<T extends Entity> Entity class: entity
- Network: network
- Entry node: waitingHall
- On enter: (empty)
- Speed: 10

Annotations include a blue arrow pointing from the 'waitingHall' in the 'Projects' tree to the 'networkEnter' block in the flowchart. A red arrow points from the text 'Entities enter At waiting hall' to the 'waitingHall' property in the Properties window. A green arrow points from the text 'Speed specified here' to the 'Speed' property in the Properties window.

Entities enter
At waiting hall

Speed specified
here

Reserving (Seizing) a Procedure Room

The screenshot displays the AnyLogic University software interface, used for modeling and simulation. The main workspace shows a floor plan of a hospital with a network flow diagram overlaid. The flow starts at a 'source' (yellow circle), moves through 'networkEnter' (blue star) to a 'networkSeize' component (blue box). From there, it goes to 'moveToProcRoom' (blue star), then to a 'procedure' (yellow circle with a clock). The flow then proceeds to 'release' (blue star), 'moveToExit' (blue star), 'networkExit' (blue star), and finally to a 'sink' (yellow circle with an 'X').

The left sidebar shows a 'Projects' tree with 'Embedded Objects' and 'Connectors'. Under 'Connectors', there is a 'Presentation' folder containing an 'image' folder and a 'networkGroup' folder. The 'networkGroup' folder contains several components: 'waitingHall', 'staffRoom', 'storageRoom', 'procRoom1', 'procRoom2', 'exit', 'rectangle', 'rectangle1', 'rectangle3', 'rectangle4', 'polyline', 'polyline1', 'polyline4', 'polyline5', 'polyline6', 'polyline7', and 'procRoom3'. The 'procRoom3' component is highlighted.

The bottom panel shows the 'Properties' window for the 'networkSeize - NetworkSeize' component. The 'General' tab is active, showing the following settings:

- Name: networkSeize
- Show name:
- Ignore:
- Public:
- Show at runtime:
- Create presentation:
- Type: NetworkSeize<T extends Entity>
- Entity class: Entity
- List of resources { pool1, ... }^D: {procRoom}
- On enter^D:
- On exit^D:
- Queue capacity: 100
- Maximum queue capacity:

The status bar at the bottom indicates 'Selection' and 'X=573, Y=105'.

Moving to the Reserved Room

The screenshot displays the AnyLogic University software interface. The main workspace is divided into two parts: a floor plan on the left and a network flow diagram on the right. The floor plan shows a room layout with various rooms and corridors. The network diagram illustrates the flow of entities through the system, starting from a source, passing through networkEnter, networkSeize, and a procedure named moveToProcRoom, then through release, moveToExit, networkExit, and finally reaching a sink.

The Properties panel at the bottom shows the configuration for the selected 'moveToProcRoom - NetworkMoveTo' element:

- Name:** moveToProcRoom
- Show name
- Ignore
- Public
- Show at runtime
-
- Type:** NetworkMoveTo<T extends Entity>
- Entity class:** Entity
- Destination is:** Specified node Seized resource unit
- Resource:** procRoom
- On enter:** [Empty field]
- On exit:** [Empty field]

The status bar at the bottom indicates the current selection is at X=482, Y=188.

Releasing the Room

The screenshot displays the AnyLogic University software interface, showing a simulation model of a room layout and a network flow diagram. The interface is divided into several panels:

- Projects Panel (Left):** Shows a tree structure of embedded objects, including connectors, presentation elements, and a networkGroup containing various rooms (waitingHall, staffRoom, storageRoom, procRoom1, procRoom2, exit) and geometric shapes (rectangle, polyline).
- Main Workspace (Center):** Displays a floor plan of a room on the left and a network flow diagram on the right. The network diagram includes nodes for source, networkEnter, networkSeize, moveToProcRoom, procedure, release, moveToExit, networkExit, and sink. A blue box highlights the 'release' node.
- Properties Panel (Bottom):** Shows the configuration for the selected 'release' entity. The configuration includes:
 - General:** Name: release, Show name checked, Ignore unchecked, Public unchecked, Show at runtime checked, Create presentation button.
 - Type:** NetworkRelease<T extends Entity>, Entity class: Entity.
 - Release:** Specified resources unchecked, All seized resources checked.
 - Moving resources:** Return to home location checked, Stay where they are unchecked.
 - On enter:** Empty text field.
 - On exit:** Empty text field.